MCFVB-CBIR: Multi-Color Feature Vector Based CBIR Using Color and Geometrical Features

Varsha Rani Patidar[1], Subhash Chandra Jat[2]
[12]Rajasthan College of Engineering for Women, Jaipur

Abstract- In computer vision and image processing digital word illustration offers many advantages for processing and distributing image and other types of understanding. A single color image have more information of the different situations and object scenario and each color have own characteristics. In the color image processing color information extraction is the key issue. CBIR finds and show images alike to one given as query image and another are similar important images. In the earlier work, most of the researchers used low level features but there are some challenges like distortion of image, color imbalance, diffraction & refraction, Semantic gap between low level features etc.To beat this issue, we proposed a new method; our proposed method works on color features of the image by combining both color histogram and CCV method and object contour with feature vector entries and adaptive resolution of the feature vectors. Then, in the retrieval algorithm, every index is compared to find some similar images to the specified query image.

Keyword- CCV, color histogram, geometrical information, CBIR.

I. INTRODUCTION

Research in the field of Content Based Visual Information Retrieval began in mid-1990's [1, 5] and is probably going to keep amid the main decade of the 21st century [7]. Since 1997, the quantity of research publications on the procedures of visual data extraction, association, ordering, client query and interaction, and database administration has expanded tremendously. Also, countless and business retrievalsystems have been produced by colleges, government associations, organizations, and doctor's facilities.

With the advancement of Internet and the user-friendliness of efficient images catching gadgets, for example, automated cameras, image scanners and high limit public networks, cheap storage capacity; the measure of advanced image collection is expanding quickly. Question by content has been an extremely dynamic research field with numerous systems proposed by modern, medicinal, and scholarly terms. CBIRSs can be divided into two types: Narrow Domain Applications and Broad Domain Applications [1]. Some applications are Crime anticipation, Architectural and building structure, Art collections, Web searching, Intellectual property and Museum the executives etc.

There are essentially three different ways of retrieving recently stored mixed media information [17]:

Free browsing: Clients peruse through an image collection, video, and sound files, and stop when they locate the desired information.

Text based: Text and literary data is included either manually or using programmed instruments to the varying media documents amid the indexing stage [8].

Content based: client search through the sight and sound archive giving data about the real content of the image, sound, and video clip.

The first two methods have some drawbacks so with the end goal to conquer the wasteful aspects and constraints of text based image retrieval of recently explained visual information,many researchers began to explore conceivable methods for retrieving visual data dependent on its content[3].

II. MOTIVATION AND OBJECTIVE

The amounts of collections of digital images have grown continuously during this period by online users, for example, in web applications that allow adding images and digital albums [1]. Also is significant to note that the images are globally used. In this context, it is necessary the development of suitable systems to deal within actual fact these collections [6]. Issue was the complexity of image information, and this information can be interpreted in different ways, consequently bringing up the issue of how to function with the end goal to control these information and express to or build up strategies to its content. This spurred the introduction of the image retrieval area whose objective is attempted to take care of those issues.The target of this work is to create and implement new methodologies of CBIR for large databases using low- and high level features, for example, color, shape and texture and geometric information that will help to: Accurate and proficient retrieval of various images exceptionally compelling those are like a query image [9].

Organization: The paper is organized into the following sections. Section II deals with the literature review of few related important papers work in utilized for the dissertation work. In Section IIIproposed approach methodology. In section IV specified problems including all the solution and result analysis contains all the evaluation regarding proposed work it deals with MATLAB GUI block and guide. In last section conclusions and outcomes after applying the proposed approach on the images of the existing work as well as on new images.

III. LITRATURE REVIEW

BMI launched the primary system for CBIR [1] and later, extraordinary image retrieval methods were it dependent on spatial design, texture, color shades, and shape [1, 2]. CBIR has attracted scientists from many research fields, including Computer vision, man-made consciousness, human factors, and machine learning.

This section aims to present a to the point overview of existing approaches in the CBIR area. To find a good image signature using suitable image features is far from easy. We include extraction and determination procedures received in retrieval system; a strategy that uses the visual content of a still image to look for comparable images in substantial scale image databases, as per a client's interest.

Flickner et al. proposed IBM's QBIC system which is a standout amongst the most outstanding CBIR systems and has a long history. Another quite recent product is Google Image search, which is a feature in the Google search engine "Search by Image". It is optimized to work well for content that is reasonably well depicted on the web, and thus, will in all likelihood give more applicable outcomes for famous landmark or paintings works of art than for more close to home images.

Pushpalatha S. Nikkam et.al. [4] There is a strategy to found a desired image from a considerable database. A layout for shape based various leveled feature matching technique for CBIR system. Shape based global feature formats. The method depends upon hierarchal information decomposition executed to distinguish boundary, introductions and dataset images shape.

Ekta Gupta et.al. [5] Present that CBIR uses visual image content for instance global color features, shape include, texture feature, and nearby features spatial area present to demonstrate and image data. CBIR method joins global and neighborhood features. In this paper SVM classifier used.

Kamlesh Kumar et.al. [6] Present CBIR system which has extra consideration from its generic to specific use. CBIR depends on visual low-Level feature extraction these are based on color shades, texture, shape and spatial design. They have used LBP technique for texture and image analysis and furthermore it is contrasted and normal RGB color image descriptor technique. And afterward a correlative feature extraction strategy applying on RGB and LBP texture method has been proposed for CBIR. Euclidean distance is use as comparability measure for finding similar images in the database.

Shu Wang et.al. [10] Proposed an orderly method to reduce appearance SBIR gap. In this paper, outlines and extracted boundary (edge) are treated as set of contour segments, establishing the system for better draw edge description and noise affect reduction from another point. Entity edge selection algorithm is used to reduce the effect of noisy edges. In this paper, spatial constraints and coherent constraint are proposed to filter the false matches that reduce retrieval rate.

Xiang-Yang Wang, et. al.[11]: proposed a novel CBIRS, using color and texture information both. At first, the image is changed from RGB space to adversary chromaticity space and the distinction of the color content of an image is imprisoned by using Zernike chromaticity distribution moments from the chromaticity space.

WichianPremchaiswadi et al. [12] present Auto color correlationsystem to help the supply of image element to catch neighborhood spatial relationship among totally color correlogram technique. At the point when an ACC is defined by using a binary matrix, it doesn't downsize the measure of bins. Hence, a decimal transformation of the binary stream in each row of the matrix is introduced in his work.

IV. PROPOSED METHODOLOGY

In this section, our focus is on feature extraction and stored as feature vector. There are some issues in feature extraction. The accomplishment of solution for issue essentially relies upon the stability and adaptability of the image features used and the qualities of the index similarity calculation method used for comparing the image features. This section presents the proposed method of image retrieval. We use color based and geometric based features for the retrieving image. Color feature extraction is finished by color moment, structure of color histogram and CCV. Here we use Corel dataset, in these dataset different categories of images and extract the features from the images of each classes separately.

A. Proposed Algorithm

Step 1: Input an image for image retrieval.

Step 2: Store RGB color component values in three different arrays, discover the image histogram of every part and compute the whole of 256different bin of RGB component.

Step 3:Color quantization is done usingcolor histogram by allocating 8 level each to hue, saturation and value to give a quantized HSV space with 8^3=512 histogram bins.

Step 4: Compute 4 different array from original array which is created after reading the image. In the first array, 1st row value of the original image will be stored as last row. In second array, last row value of the original image will be stored as first row.

Step 5: Calculate the normalization of 5 image array.

Step 6: Then make a matrix which is combination of RGB (R+G), (R +B), (B +G), White and Black value of an image and they are initializes as default value.

Step 7: Now we measure the magnitude of difference from the average matrix which is get in and the value said in step 6 for each 8 distinct element.

Step 8: Then we calculate the threshold value which will give best result.

Step 9: Using the threshold value calculate how many Red channel connected pixel we get and we called it as a Red Coherent (Re), and find also how many pixel do not contain any red information and we called it as a Red incoherent (Ri). We continue this process for all 8 different color information said in step6.

Step 10: Finally we get different Coherent and Incoherent value from the image and store it in a previously created array.

Step 11: Repeat step 1 to 10 for each image in the database.

Step 12: Calculate the similarity matrix of query image and the image present in the database.

Step 13: Retrieve the images.

B. System Architecture

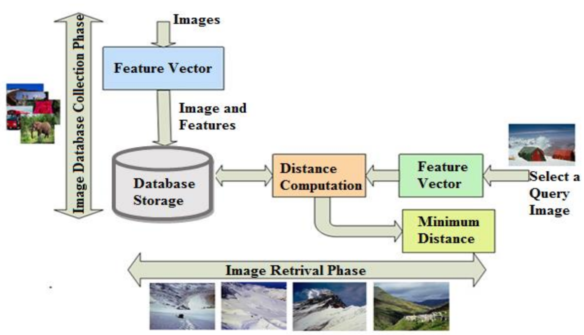

Fig.1: Block Diagram of Image Retrieval System

C. Features for image retrieval

i. Feature Extraction using color information

Colorimage have no of colors and each color have own property, which normally used visual features in CBIRS. The objective of color ordering is to recover every one of the images whose color creations are like the color arrangement of the query image. Color data is spoken to as focuses in three dimensional color spaces, for example, RGB, HSV, L*a*b*, YIQ.

ii. Color Histogram

It describes the extent of various colors in the extent of the entire image, and does not think about which the spatial position of each color. This algorithm can be divided into three stages: segment of the color space into cells, relationship of every cell to a histogram bin, and checking of the quantity of image pixels of every cell and putting away this tally in the comparing histogram bin. This procedure is called color quantization. The impact of color quantization on the execution of image retrieval has been accounted for by numerous authors [13, 14]. In our method we consider all images in different size each image have different height and width. We discretize the colors pace of the image with the end goal that there are n distinct colors. In our technique we consider 8-bit image. In this way, there will be 256 distinct bins of colors. At that point we take a summation of it and locate the mean and standard deviation from the color histogram. At long last it put away in a 1D array. This value is determined for each image in the database.

$$\mathrm{H} = \{\mathrm{H}[0], \mathrm{H}[1], \mathrm{H}[2], \mathrm{H}[3], \ldots\ldots\ldots \mathrm{H}[i], \ldots\ldots, \mathrm{H}[n]\} (3.1)$$

The normalized color histogram H' is given as:

$$\mathrm{H}' = \{\mathrm{H}'[0], \mathrm{H}'[1], \mathrm{H}'[2], \mathrm{H}'[3], \ldots\ldots\ldots \mathrm{H}'[i], \ldots\ldots\ldots, \mathrm{H}'[n]\} \quad (3.2)$$

Where$\mathrm{H}' = \frac{H[i]}{P}$, is the total number of pixels of an image

Fig.2: Color histogram analysis of Bus image

iii. Color moment

Color moments are mainly used for color indexing purposes as features in imageretrieval applications with the end goal to compare at how comparable two images depend on color. Usually one image is compared to a database of digital images with pre-computed features in order to find and retrieve a similar image. Every comparison between images results in a similarity score, and the lower this score is the more identical the two images should be. Colormoment is scaling and rotation invariant. It is usually the case that only the first three color moments are used as features in image retrieval applications as nearly all of the color distribution information is contained in the low-order moments [4]. In addition, due to the color distribution information for the most part concentrated in the low-order moments. Three low-order moments of colors communicated mathematically as:

Mean: The first color moment can be deciphered as the normal color in the image, and it tends to be determined by using the accompanying equation (3.3):

$$\mu_i = \frac{1}{N}\sum_{j=i}^{N} p_{ij} (3.3)$$

Where N is the quantity of pixels in the image and pij is the value of the jth pixel of the image at the ithcolor channel.

Standard Deviation: The second color moment is the standard deviation, which is obtained by taking the square root of the variance of the color distribution.It can be calculate as (3.4):

$$\sigma_i = \left(\frac{1}{N}\sum_{j=i}^{N}\left(p_{ij}-\mu_i\right)^2\right)^{\frac{1}{2}} (3.4)$$

Where $\mu_i$is the mean value, or first color moment, for the ith color channel of the image.

Skewness: It measures how asymmetric the color distribution is, and in this way it gives data about the shape of the color distribution. Skewness can be computed with this equation (3.5):

$$S_i = \left(\frac{1}{N}\sum_{j=i}^{N}\left(p_{ij}-\mu_i\right)^3\right)^{\frac{1}{3}} (4.5)$$

Here $P_{ij}$is the image of the j^{th} pixel in the i^{th} color element. Thus, the total moment of color image has nine components each moment associated with three color components.

Color Indexing

A function of the comparability between two image spreads is characterized as the aggregate of the weighted difference between the moments of the two distributions[11]. It can be designed as:

$$D_{moment}(H,I) = \sum_{i=1}^{c\square} w_{i1}|\mu_i^1 - \mu_i^2| + w_{i2}|\sigma_i^1 - \sigma_i^2| + w_{i3}|S_i^1 - S_i^2| \quad (3.6)$$

Where:

(H,I) are two image color dissemination being compared

i : is the recent channel index (R=1,G=2, B=3), ch : represents total number of channels(3)

$(\mu_i^1, \mu_i^2), (\sigma_i^1, \sigma_i^2), (S_i^1, S_i^2)$ represent the Mean, SD, Skewness respectively of the two image dissemination w_i: are the weights for each moment.

iv. Auto Color-Correlogram

Color Correlogram and color moments perform better than traditional color histograms used for CBIR.Correlogram can be put away as a table indexed by sets of colors (i, j) where dth entry demonstrates the probability of finding a pixel j from pixel i at distance d. Though a colorcorrelogram can be put away as a table listed by color i where d-th entry demonstrates the probability of finding a pixel i from a similar pixel at distance d. Henceforth colorcorrelogram shows the spatial connection between identical colors only [15].

v. Color Coherence Vector

There exist many CBIR systems, which retrieve images by color histogram. Yet, just histogram based strategy can't give best outcomes if two images have same color dispersion.

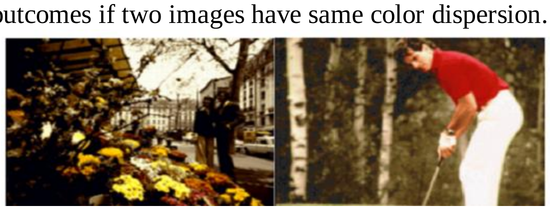

Fig.3: Two images with similar color histograms

For example, the images shown in figure 3.3 have comparative color histograms, regardless of their rather different appearances. The color red shows up in the two images in roughly similar amounts. The color red shows up in the two images in roughly similar amountsIn the left image the red pixels (from the flowers) are commonly dispersed, while in the right image the red pixels (from the golfer's shirt) shape a solitary coherent region.

Hence we Consider Color Coherence Vector additionally as a color feature extraction technique. We characterize a color's coherence as how much pixels of that color are individuals from huge likewise colored regions. We allude to these significant regions as coherent regions, and see that they are of huge significance in describing images.

Our second system on feature extraction is color coherence vector esteem examination. In CCV each histogram container is apportioned into two sorts: coherent and incoherent. All pixels which casually fall into a comparative colored region are called coherent type. Else it calls incoherent in sort. We decide the pixel groups by computing associated components. In this strategy some locale is created dependent on color data. Here, we measure Red, Green, Blue, Red+ Green, Red+ Blue, Blue+ Green, White and Black Coherent-Incoherent type and store it in a 1D array. This value is calculated for every image in the database.

At last we join each all the three extracted component in one matrix and repeated the procedure for each image. Then we apply to similarity measurement between the images.

vi. CCV's Computation

The underlying stage in computing a CCV is like the computation of a color histogram. We first degrade/ blur the imageslightly by replacing pixel values with the normalizevalue in a small nearby neighborhood.

This disposes of small variations between neighboring pixels. We then discretize the color space, with the end goal that there are just n distinct colors in the image. The subsequent stage is to group the pixels inside a given color bucket as either coherent or incoherent. A coherent pixel is a piece of a large group of pixels of a similar color, while a mixed up pixel isn't. We decide the pixel groups by computing associated components.

An associated component C is a maximal arrangement of pixels with the end goal that for any two pixels p, p0 ∈ C, there is a way in C among p and p0.

We view two pixels as adjoining in the event that one pixel is among the eight nearest neighbors of the other; in other words, we include corner to corner neighbors. Note that we just register associated components inside a given discretized color bucket. This effectively portions the image dependent on the discretized color space. Associated components can be processed in linear time. At the point when this is completed, each pixel will have a place with precisely one associated component.

We arrange pixels as either coherent on incoherent depending on the size in pixels of its related segment. A pixel is coherent if the extent of its associated componentexceeds a settled value τ; something else, the pixel is incoherent. For a given discretized color, a bit of the pixels with that color will be coherent and some will be incoherent. Allow us to call the quantity of incoherent pixels of the jth discretized color αj and the quantity of incoherent pixels βj. Plainly, the aggregate number of pixels with that color is αj + βj, thus a color histogram would summarize an image as:

$$< \alpha_1 + \beta_1, \dots, \alpha_n + \beta_n > \quad (3.7)$$

As a substitute, for each color we compute the pair

$$(\alpha_j + \beta_j) \quad (3.8)$$

Which we will call the coherence pair for the jth color. The color coherence vector for the image involves

$$< (\alpha_1, \beta_1), \dots, (\alpha_n, \beta_n) > \quad (3.9)$$

This is a vector of coherence pair, one for each discretized color.

vii. The Combination of Color Histogram and CCV

Since color histogram in expressing the global color distribution while CCV is more sensitive to spatial information of an image, we can combine them to get the best of them. In this approach, RGB model is used and the cube is subdivided into 216 sub-cubes uniformly. The mathematical expression of the algorithm of color quantization is

$$I = \left\lfloor \frac{R}{N} \right\rfloor \times n^2 + \left\lfloor \frac{G}{N} \right\rfloor \times n^1 + \left\lfloor \frac{B}{N} \right\rfloor \times n^0 \quad (3.9)$$

Where I is the index of the quantized color, n is the number of portions of each edge of the image. After quantization every pixel in the image is related with a color record I and the pixels have a similar index are thought as having a similar color. Quantization is vital, before quantization there are 2^{24} number of colors. It is impossible, for any reason, to manage this number of colors and we don't need that many either. Next the color histogram is processed. Since the calculation of color histogram is straightforward, I will skip its formulation here. RGB 3D square after quantization, N is the progression width and N = 256/n, and R, G, and B are the R, G, and B estimations of the RGB image. The most extreme number of colors after quantization is

$$\left(\left\lfloor \frac{256}{n} \right\rfloor + 1\right)^3 \quad (3.10)$$

Next we compute the CCV for the images and store the outcome in a binary file. Using this pre-computation optimization, we have no need reason to construct the database each time when the program is begun and this will save a huge amount of time. The meaning of coherency is that if a pixel in the image interfaces with over 1% pixels in the image in the sense of 4-connectivity, at that point the pixel is believed to be coherent else it is incoherent. The algorithm for checking coherency can be executed using recursion naturally yet recursion wastes memory and it is ease back because of frequent function calls so we'd recreate the recursive procedure using a stack. Moreover, in the event that a pixel is coherent, every one of the pixels that are 4-connective to it are additionally reasonable and in the event that it is disjointed, every one of the pixels that are 4-connective to it are incoherent as well. Along these lines as opposed to checking coherency for every pixel, we can check coherency for an arrangement of pixels in only one pass and subsequently maintaining a strategic distance from repetitive calculation. After all theseenhancement, the calculation is exceptionally productive with the end goal that the time required for each inquiry is short of what one moment. That is, you will see the outcomes with about no delay. To make things finish, the CCV is a variety of 2-tuples, as

$$\mathrm{CCV} = [< a_0, b_0 >, < a_i, b_i >, \dots < a_m, b_m >] \quad (3.11)$$

Where an i is the number of coherent pixels of color i, bi is the number of incoherent pixels of color i, and m the number of colors used. To depict the similitude of images, we require some standard, for example, distances: the L1 distance, the Euclidean distance, the L∞ distance, and so on. What's more, we choose L1 distance since it gives the best performance. The formula for computing distance is equation (4.12) in similarity part.

Where D12 is the L1 distance of images 1 and 2, h1 is the color histogram of the query image, h2 is the color histogram of a subjective image in the image database, CCV1 is the CCV of the query image and CCV2 is the CCV of a self-assertive image in the database. The function sum () includes every one of the sections together and thus these lines its return value is a scalar. So in the event that Dxy<Dxz ,image y is viewed as more like image x than image z is. At long last, the one thousand distances are arranged in ascending order and showed all together.

viii. Similarity computation

The likeness between two color histograms can be performed by figuring the L1, L2 or weighted Euclidean divisions and by registering their convergence point. The second step includes coordinating these features to yield an outcome that is outwardly similar. Basic idea behind CBIR is that, while building an image database, incorporate vectors from images is to be expelled and after that store the vectors in another database for quite a while later.At the point when given a query image its component vectors are computed. On the off chance that the separation between feature vectors of the query image and image in the database is little enough, the relating image in the database is to be considered as a match to the query. The search is generally based on closeness instead of on correct match and the retrieval results are then positioned as needs be to a comparability file Similarity estimation furthermore, coordinating involves a significant job in CBIR algorithms. These algorithm search image database to discover images like a given query with the goal that they ought to have the capacity to assess the measure of similarities between the images. To measure the similarity, the direct Euclidean distance between the image in the database and the query image is given by,

$$D_{12} = .5 \times \text{sum}(|h_1 - h_2|) + .5 \times \text{sum}(|CCV_1 - CCV_2|) \qquad (3.12)$$

V. EVALUATION PROTOCOL AND RESULT ANALYSIS

Evaluation of proposed work already discussed in the previous chapter. We introduce the database we select to test our system, and compare our system results with other existing CBIRS that most of them use the same image database.

We use WANG database, it [16] is an image database that the images are manually selected from the COREL gallery image database for performing experiment which contains natural scenery images such as: photos, web images, animations, video and clips. The Corel image database contains countless of different content, ranging from creatures and open air sports to normal scenes.

These images are pre-ordered into various classifications of size by space experts. We have chosen 1000 images from the database. The images are separated into 10 classes. Each class contains 100 images. It is generally used for testing CBIR frameworks. In figure 5.1 we showed sample images from each class.

Fig.4: Sample images from WANG database [18]

Here, we test our proposed algorithm and show the results. In order to check retrieval effectiveness of the proposed system, we have to test it by selecting some images randomly and retrieve some relevant images.

One of the most important components of image retrieval system is user interaction (UI) with the system. For the sake of simplicity and running we decide our UI as a MATLAB GUI.

At very first step in the processing of image indexing, using MATLAB open command window >> start with run a main file is saved with the name “gui”, The first screen which appears after the command as shown in the figure 4.2 which has three basic functionality, first one to select query image and the second one is to relevance results and final result.txt file.

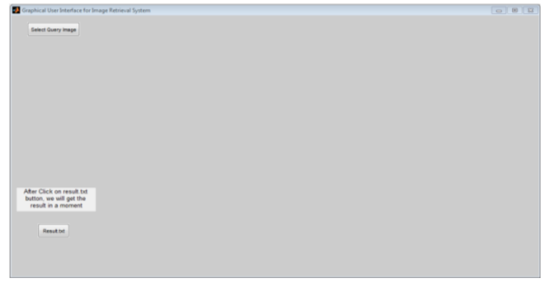

Fig.5: Graphical User Interface window

Commonly used query formations are: categorybrowsing, Category browsing is to browse through the database according to the category of the image. By this specification we can choose a query image and then select a query image form database shown in figure 4.3.

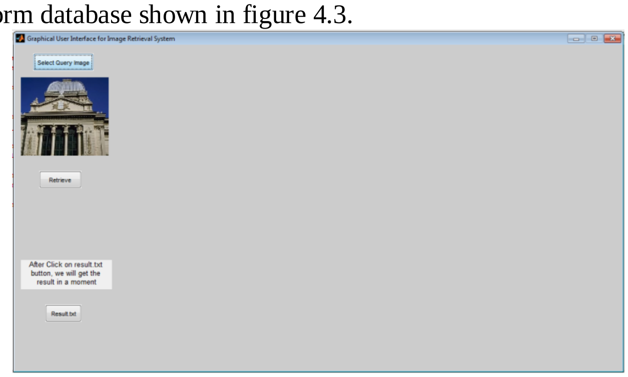

Fig.5: Selected image shown on axis handler

The first test is selecting a random image from the building class. We submit the image to the system and retrieve the top 25 images that are similar to the query image. As can be seen from figure 4.4 we select the query image from the building class randomly and retrieve the most top 25 images that are like to the query image.Notice that the retrieved images are having a place with a similar class of the query image, and every one of the images are comparative in the color feature. In this way, the framework recovers them appropriately.

Fig.6: Retrieve image of class building

If we look to Figure deeply, we notice that the last two three images retrieved by the system has an object with different color (the blue or gray color). Although the image Asian and beach with a strange color from other images, the system retrieves it and assigns it to the Building class. This is true because the major color for the image is similar to the query image, so the system retrieves it.Here we have 10 classes but some classes are shown in figure.

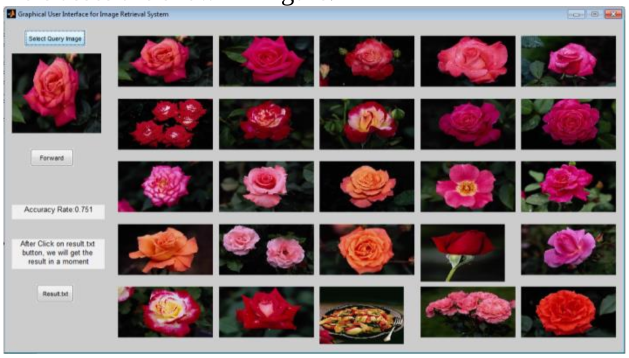

Fig.7: Retrieve image of class flower

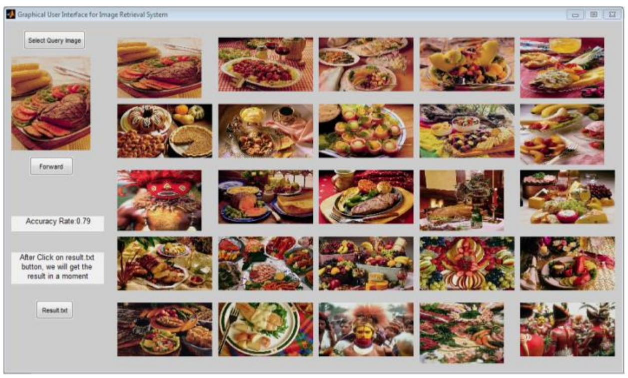

Fig.8: Retrieve image of class food

a. Performance Evaluation matrices

Generally, calculate the two performance evaluation metrics that are Precision and Recall. Precision (P) measures the accuracy of the retrieval and Recall (R) measures the robustness. The metrics are defined as:

$$P = \frac{Number\ of\ relevant\ images\ retrieved}{Total\ number\ of\ images\ retrieved} \quad (4.1)$$

$$R = \frac{Number\ of\ relevant\ images\ retrieved}{Total\ number\ of\ relevent\ images\ in\ the\ dataset} \quad (4.2)$$

Accuracy

An accuracy of the system can be defined as the half the fraction sum of precision and recall values, which can be expressed as,

$$Accuracy = \frac{(P + R)}{2} \quad (4.3)$$

Matrices values are commonly represented in a precision-recall graph, R→P(R) brief (R, P(R)) pairs for varying numbers of retrieved images.

Finally, we compare our proposed system with other existing systems and show the efficiency of the proposed system. To test the system, we select 250 different images from WANG database and from different classes (25 images from each class).

b. Overall System Analysis

Table 4.1: Matrix for different classes

Class	Food	Mountain	Africa	Beach	Building	Bus	Elephant	Flower	Horse	Dianosour
Food	89	0	6	0	0	0	4	1	0	0
Mountain	0	78	3	12	0	5	2	0	0	0
Africa	9	3	82	0	2	4	0	0	0	0
Beach	1	6	0	78	8	7	0	0	0	0
Building	0	0	7	4	78	5	6	0	0	0
Bus	5	0	7	0	4	84	0	0	0	0
Elephant	0	0	6	0	0	0	76	2	16	0
Flower	4	0	0	0	0	6	0	87	3	0
Horse	1	0	4	1	0	1	7	6	80	0
Dianosour	0	0	0	0	0	0	0	0	0	100

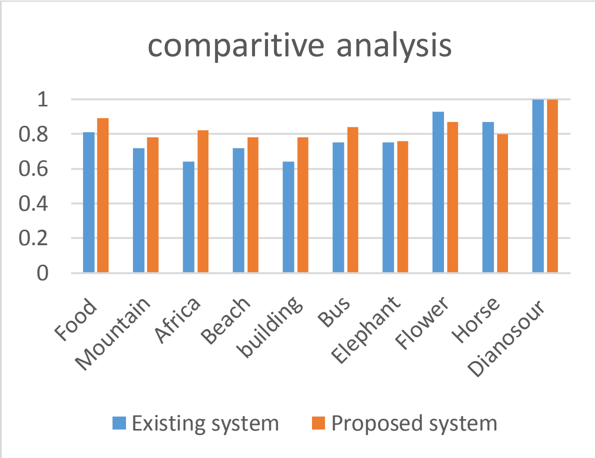

Fig.9: Precision and recall compression

c. Comparison of proposed work with excising systems

Table 4.2: Comparison along with proposed scheme and the existing scheme in terms of average precision rate (mean value +- standard deviation) for WANG database

Method / Category	**Silakari et.al [19]**	**CTDCI RS[21]**	**Poursistani et.al. [20]**	**Proposed**
Food	0.22	0.69	0.74	0.89
Mountain	0.43	0.51	0.56	0.78
Africa	0.33	0.56	0.7	0.82
Beach	0.42	0.54	0.44	0.78
Building	0.79	0.61	0.71	0.78
Bus	0.45	0.89	0.76	0.84
Elephant	0.45	0.57	0.64	0.76
Flower	0.94	0.9	0.92	0.87
Horse	0.59	0.78	0.95	0.8
Dianosour	0.97	0.98	1.00	1.00

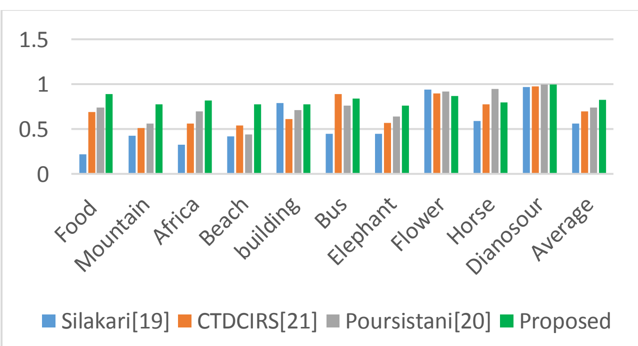

Fig.10: Comparison of precision of the proposed system with existing systems

d. CONCLUSION AND FUTURE WORK

Our contributions and work conclusion with some recommendation and future works.CBIR is an active research area since the 1990s. In our work, we perform extraction on color image and extract color features for the matching. Because of all images in our world are color images. So, color feature is one of the nearly all features that can be taken into description when developing a CBIRS. We used it to evaluate the performance of our system by calculating the Precision & Recall metrics and overall system accuracy. We also compared our proposed system with other existing systems that use the same WANG database we have used for system evaluation. The comparison shows that our system outperforms the other systems and the results are satisfactory.
Here is a drawback that the only color feature is not enough to represent the image and use for similarity matching. From the image retrieve result, we notice that some retrieved images are not related to the selected image. These images based only dominant color so we can overcome this limitation by using more than one feature to represent the image

VI. REFERENCES

[1]. Datta R, Joshi D, Li J, Wang JZ," Image retrieval: Ideas, influences, and trends of the new age", ACM Comput Surv 40(2), 2008.
[2]. Ashraf R, Bashir K, Mahmood T, "Content-based image retrieval by exploring bandletized regions through support vector machines", J Inf Sci Eng 32(2):245–269, 2016.
[3]. Mvhammad Hammad me mon!, jian-ping li!, imran memon, riaz ahmed shaikw and farman alimangi," efficient object identification and multiple regions of interest using cbir based on relative locations and matching regions", pp: 247-250, IEEE 2015.
[4]. Pushpalatha S. Nikkam, Dr. Nagaratna P. Hegde and Dr. B. Eswar Reddy," Decomposition-Based Shape Template Matching for CBIR System", International Conference on Computational Intelligence and Computing Research, IEEE 2015.
[5]. Ekta Gupta and Rajendra Singh Kushwah," Combination of Global and Local Features using DWT with SVM for CBIR", IEEE 2015.
[6]. Kamlesh Kumar, Jian-Ping Li And Zain-Ul-Abidin," Complementary Feature Extraction Approach In CBIR", Pp: 192-197, IEEE 2015.
[7]. Radu Andrei Stefan, Ildikó-Angelica Szöke and Stefan Holban," Hierarchical clustering techniques and classification applied in Content Based Image Retrieval (CBIR)", 10th Jubilee IEEE International Symposium on Applied Computational Intelligence and Informatics , Timişoara, Romania, pp: 147-152, May 21-23, 2015
[8]. KattaSugamya,SureshPabboju, Dr.A.VinayaBabu, "A CbirClassification Using Support Vector Machines" 978-1-4673-8810-8/16/$31.00 ©IEEE 2016.
[9]. Syntyche Gbèhounou, François Lecellier, Christine Fernandez-Maloigne, "Evaluation of local and global descriptors for emotional impact recognition". Elsevier 2016.

[10]. Shu Wang, Jian Zhang, Senior Member, IEEE, Tony X. Han, Member, IEEE, and Zhenjiang Miao, Member, IEEE, "Sketch-Based Image Retrieval Through Hypothesis-Driven Object Boundary Selection With HLR Descriptor", vol. 17, no. 7, pp 1045-1057, july 2015.

[11]. Xiang-Yang Wang, Hong-Ying Yang, Dong-Ming Li "A new content-based image retrieval technique using color and texture information", Computers & Electrical Engineering, Pages 746-761, Volume 39, Issue 3, April 2013.

[12]. Wichian Premchaiswadi, Anucha Tungkasthan, "A Compact Auto Color Correlation using Binary Coding Stream for Image Retrieval", Proceedings of the 15th World Scientific and Engineering Academy and Society (WSEAS) International conference on Computer, Recent Researches in Computer Science. pp 430-436.2011.

[13]. X. Wan and C. C. J. Kuo, "A new approach to image retrieval with hierarchical color clustering", IEEE transactions on circuits and systems for video technology, Vol. 8, no. 5, 1998.

[14]. X. Wan and C. C. Kuo, "Image retrieval with multiresolution color space quantization", In Electron Imaging and Multimedia System, 1996.

[15]. Dipankar Hazra. Retrieval of color image using color correlogram and wavelet filters. Proc. of International Conference on Advances in Computer Engineering, 2011.

[16]. J.Z. Wang, "Wang Database," [Online], Available at: http://wang.ist.psu.edu/, last visited August 12th 2010.

[17]. S. Silakari, M. Motwani, and M. Maheshwari, "Color image clustering using block truncation algorithm," Int. Jour. of Comp. Science Issues, vol. 4, no. 2, 2009.

[18]. Database:http://wang.ist.psu.edu/docs/related/test1.rar.

[19]. S. Silakari, M. Motwani, and M. Maheshwari, "Color image clustering using block truncation algorithm," Int. Jour. of Comp. Science Issues, vol. 4, no. 2, 2009.

[20]. P. Poursistani, H. Nezamabadi-pour, R. A. Moghadam, and M. Saeed, "Image indexing and retrieval in JPEG compressed domain based on vector quantization," Math. and Comp. Modeling, 2011.

[21]. M. Babu Rao, B. Prabhakara Rao, and A. Govardhan, "CTDCIRS: Content based Image Retrieval System based on Dominant Color and Texture Features," International Journal of Computer Applications (0975 – 8887), Volume 18– No.6, March 2011.